

### REMARKS/ARGUMENTS

Applicant has herein amended claims 3, 4, 7, 8, and 12. Claims 1-14 will still be pending in the application after entry of this amendment. No additional fee is due. Applicant hereby requests reconsideration in view of the foregoing amendments and the remarks made below.

A brief telephonic interview took place between the Examiner and the undersigned attorney on December 13, 2005, in which the claim objection due to the use of the phrase "hypertext transmission protocol" was discussed. During this interview, the undersigned attorney proposed to amend the objected-to claims as recommended by the Examiner, but also to make a parallel correction to the specification. The Examiner agreed that this change would be acceptable and would not constitute new matter. Applicant thanks the Examiner for his willingness to discuss the matter and move this application forward.

The Examiner has rejected claims 1, 2, 5, 6, 9-11, 13, and 14 under 35 U.S.C. § 112, second paragraph, for failing to particularly point out and distinctly claim the subject matter of the invention. The Examiner bases this rejection on the use of the term "object" in these claims, saying that it is unclear what an object is. Applicant submits that the term object is a standard term that is well-known in the computer programming arts, and thus requires no definition, but should be given its ordinary meaning in the art. Object-oriented programming techniques have been known for many years. The *McGraw Hill Dictionary of Scientific and Technical Terms, Fifth Edition*, published in 1994, defines object oriented programming as: "A computer programming methodology that focuses on data rather than processes, with programs composed of self-sufficient modules (objects) containing all the information needed to manipulate a data structure. The book, *Object-Oriented Technology: A Manager's Guide*, by David A. Taylor, P.h.D., published in 1990 defines object on page 16 as, "a software 'package' that contains a collection of related procedures and data." Page 15 of that book traces the origins of objects and object-oriented programming to the Simula language developed in the 1960's. Indeed, at least some of the art that has been cited in the prosecution of this application uses the term "object" as well. Applicant submits that the term "object" would be well-understood by those of skill in the art without an express definition being given in the specification.

The Examiner has rejected all claims under 35 U.S.C. § 102(e) as anticipated by U.S. Patent 6,256,712 to Challenger et al. ("Challenger"). In order for a claim to be anticipated, the

cited reference must teach every element of the claim, either expressly or inherently. MPEP 2131. All of Applicant's claims, either directly, or through dependency, have recitations that cannot be found in Challenger. As but one example, all of Applicant's claims recite a request and a "response that can be displayed as a combination of a dynamic protocol object and a static protocol object." Challenger, by contrast, does not discuss responding to requests. Rather Challenger discusses combining objects into larger objects as an update mechanism. Additionally, all of Applicant's claims recite the use of a cache disposed in an operating system kernel. Applicant is also at a loss to find this concept disclosed in Challenger. The portions of Challenger cited by the Examiner discuss either a proxy cache or a processor cache, neither one of which resides in a kernel. A proxy cache resides in user space and a processor cache resides inside the processor hardware. Challenger does not even mention the kernel, let alone an in-kernel cache.

Applicant believes he has addressed all of the Examiner's concerns. Reconsideration and allowance is hereby requested.

Respectfully submitted,

Date: 9 JAN 06

Telephone: (919) 286-8000  
Facsimile: (919) 286-8199



Steven B. Phillips  
Registration No. 37,911  
Moore & Van Allen PLLC  
P.O. Box 13706  
Research Triangle Park, NC 27709-3706